ARMY RESEARCH LABORATORY

Microwave Circuit Simulator for MATLAB

Romeo D. del Rosario and Daniel C. Judy

ARL-TN-184

March 2002

Contents

Figures

Army Research Laboratory
Adelphi, MD 20783-1197

ARL-TN-184 **March 2002**

Microwave Circuit Simulator for MATLAB

Romeo D. del Rosario and Daniel C. Judy
Sensors and Electron Devices Directorate

Abstract

This technical note describes a microwave circuit simulator implemented in the technical computing language, MATLAB. The simulator takes an input circuit with a known admittance matrix, performs a node swap that puts the external nodes (ports) at the upper left hand of an equivalent admittance matrix, then reduces the matrix (regardless of size) to include only the external nodes using Householder's method. The resulting 2×2 admittance matrix provides all the information to uniquely define the circuit from its external ports and facilitates straightforward calculation of Z-parameters, S-parameters, etc. (In our case, the reduced matrix is 2×2; however, the method applies to multiple port devices as well.) Although it can be used by itself, the simulator is designed with a bent toward empirical (circuit based) transistor models and may be incorporated into small and large signal transistor models.

1. Introduction

This technical note describes a microwave circuit simulator implemented in the technical computing language, MATLAB. The simulator takes an input circuit with a known admittance matrix, performs a node swap that puts the external nodes (ports) at the upper left hand of an equivalent admittance matrix, then reduces the matrix (regardless of size) to include only the external nodes using Householder's method. The resulting 2×2 admittance matrix provides all the information to uniquely define the circuit from its external ports and facilitates straightforward calculation of Z-parameters, S-parameters, etc. (In our case, the reduced matrix is 2×2; however, the method applies to multiple port devices as well.) Although it can be used by itself, the simulator is designed with a bent toward empirical (circuit based) transistor models and may be incorporated into small and large signal transistor models.

2. Algorithms

2.1 Simulator

1. Design input (circuit)
2. Assign node numbers
3. Determine self admittances (diagonals of [Y])
4. Determine trans admittances (off-diagonal elements of [Y])
5. Frequency plan
6. Calculate [Y(f)] for each frequency point
7. Apply node switch algorithm
8. Apply Householder's theorem to obtain reduced admittance matrix $[Y_{red}]$
9. Calculate other parameters e.g. [S] , [Z], etc. from $[Y_{red}]$
10. Repeat for every frequency point

2.2 Node Switch Algorithm

It is desirable to reduce, when possible, the amount of data involved in performing a circuit simulation. One approach is to reduce the admittance matrices so that only the external nodes (ports) are described. Before this can be accomplished, however, it is necessary to reconfigure the matrix, placing the external nodes in the upper left corner:

1. Assume that K and L are the nodes to be switched.
2. Start loop for rows.
3. Start loop for columns.
4. If row K (K or L) AND column K (K or L) , then $\mathbf{Y'}_{row,col} = \mathbf{Y}_{row,col}$.
5. If row K (K or L) AND column = K then $\mathbf{Y'}_{row,col} = \mathbf{Y'}_{row,L}$.
6. If row K (K or L) AND column = L , then $\mathbf{Y'}_{row,col} = \mathbf{Y'}_{row,K}$.
7. If row = K AND column = L , then $\mathbf{Y'}_{row,col} = \mathbf{Y}_{L,K}$.
8. If row = K AND column = K , then $\mathbf{Y'}_{row,col} = \mathbf{Y'}_{L,L}$.
9. If row = K AND column K (K or L), then $\mathbf{Y'}_{row,col} = \mathbf{Y'}_{L,col}$.
10. If row = L AND column K (K or L), then $\mathbf{Y'}_{row,col} = \mathbf{Y'}_{K,col}$.
11. If row = L AND column = K, then $\mathbf{Y'}_{row,col} = \mathbf{Y'}_{K,L}$.

12. If row = L AND column = L, then $\mathbf{Y'}_{row,col} = \mathbf{Y'}_{K,K}$.

13. Repeat for every frequency point.

Example

$$\begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{21} & Y_{22} & Y_{23} \\ Y_{31} & Y_{32} & Y_{33} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix}$$

becomes

$$\begin{bmatrix} I_1 \\ I_3 \\ I_2 \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{13} & Y_{12} \\ Y_{31} & Y_{33} & Y_{32} \\ Y_{21} & Y_{23} & Y_{22} \end{bmatrix} \begin{bmatrix} V_1 \\ V_3 \\ V_3 \end{bmatrix}$$

2.3 Application of Householder's Method

Assuming that a node swap has been performed in order to place the external nodes at the upper left corner of the total admittance matrix, Y_{total}, we can now reduce this matrix, regardless of initial size, to a considerably smaller one that entails only the external nodes (1 and 2 in this example for a two-port circuit.) This is achieved by virtue of Householder's method so that a matrix can be broken into submatrices as follows [3]:

$$[Y_{total}] = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} & Y_{14} & Y_{15} \\ Y_{21} & Y_{22} & Y_{23} & Y_{24} & Y_{25} \\ Y_{31} & Y_{32} & Y_{33} & Y_{34} & Y_{35} \\ Y_{41} & Y_{42} & Y_{43} & Y_{44} & Y_{45} \\ Y_{51} & Y_{52} & Y_{53} & Y_{54} & Y_{55} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{bmatrix} & \begin{bmatrix} Y_{13} & Y_{14} & Y_{15} \\ Y_{23} & Y_{24} & Y_{25} \end{bmatrix} \\ \begin{bmatrix} Y_{31} & Y_{32} \\ Y_{41} & Y_{42} \\ Y_{51} & Y_{52} \end{bmatrix} & \begin{bmatrix} Y_{33} & Y_{34} & Y_{35} \\ Y_{43} & Y_{44} & Y_{45} \\ Y_{53} & Y_{54} & Y_{55} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} Y_{ee} & Y_{ei} \\ Y_{ie} & Y_{ii} \end{bmatrix}$$

where subscripts "e" and "i" denote external and internal, respectively.

Likewise, Ohm's Law in the matrix form can be reduced.

$$I = \begin{bmatrix} I_e \\ I_i \end{bmatrix} = \begin{bmatrix} Y_{ee} & Y_{ei} \\ Y_{ie} & Y_{ii} \end{bmatrix} \begin{bmatrix} V_e \\ V_i \end{bmatrix}$$

We recall that an individual element of the admittance matrix, Y_{KL} can be calculated by shorting all nodes ≠ K or L and applying an internal current voltage, V_i, and by observing the current I_i. However, assuming a passive device, all the internal currents are zero and

$$I_e = Y_{ee}V_e + Y_{ie}V_i$$
$$O = Y_{ie}V_e + Y_{ii}V_i$$

$$V_i = -Y_{ii}^{-1}Y_{ie}V_e$$

$$\therefore I_e = \left(Y_{ee} - Y_{ie}Y_{ie}^{-1}Y_{ie}\right)V_e$$

$$Y_{reduced} = Y_{ee} - Y_{ie}Y_{ii}^{-1}Y_{ie}$$

3. Results and Comparison to ANSOFT Serenade

Figure 1. Calculated S_{21} Ansoft (circles) and MATLAB (solid).

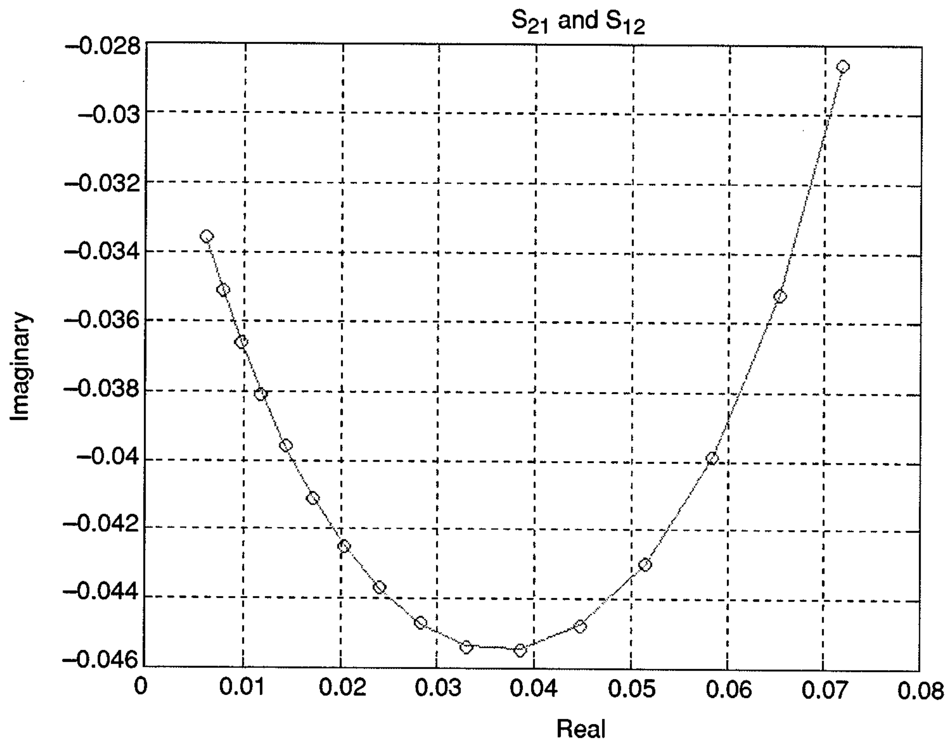


Figure 2. Calculated S_{11} Ansoft (circles) and MATLAB (solid).

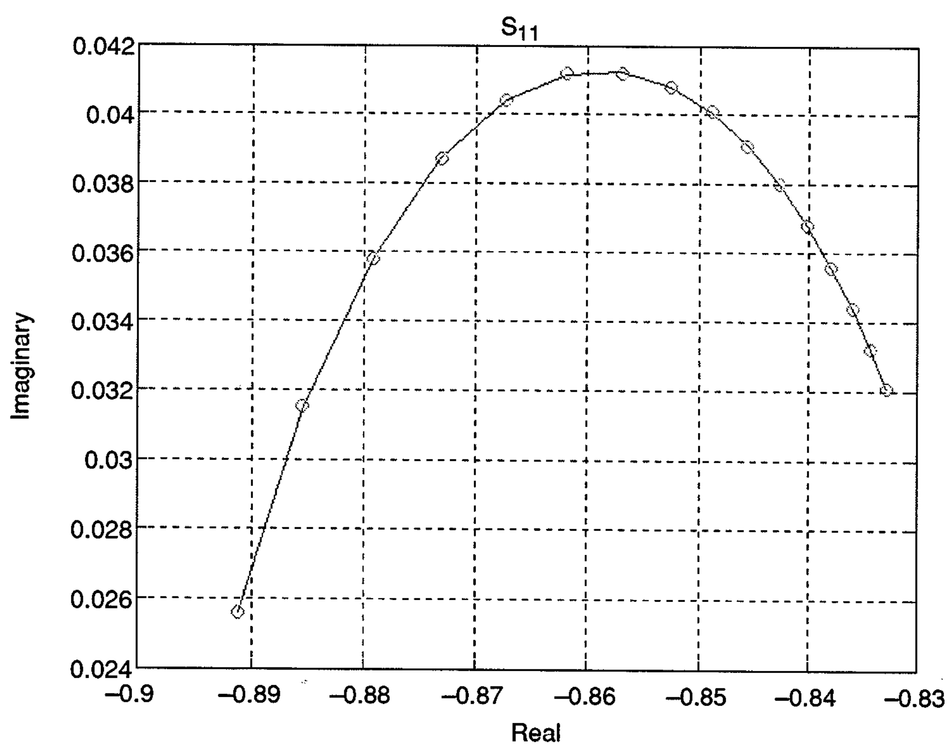


Figure 3. Calculated S_{22} Ansoft (circles) and MATLAB (solid).

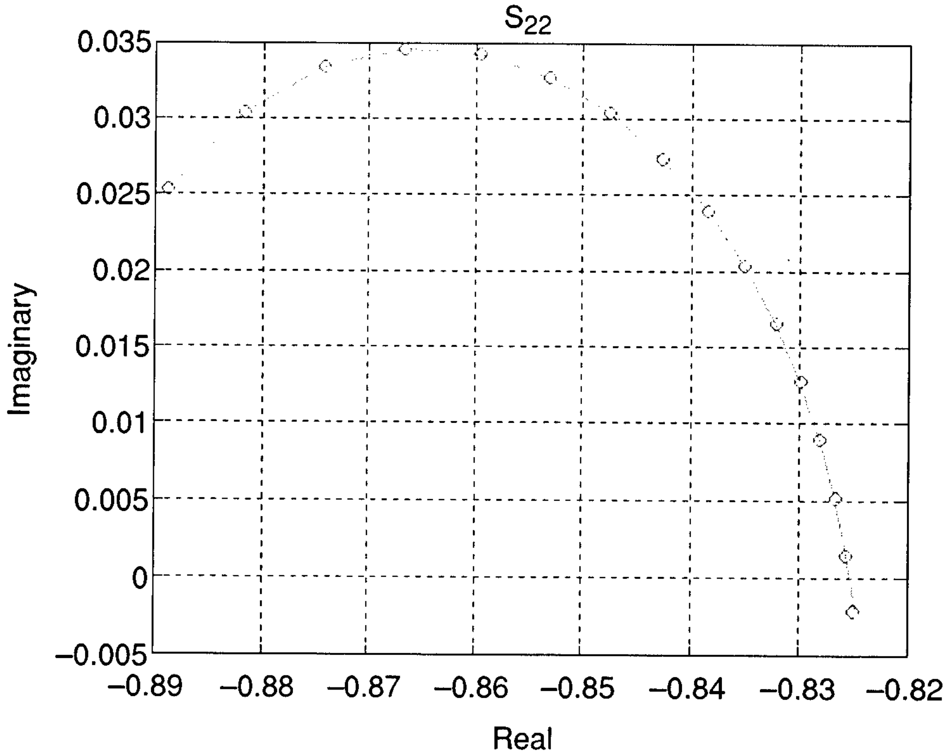
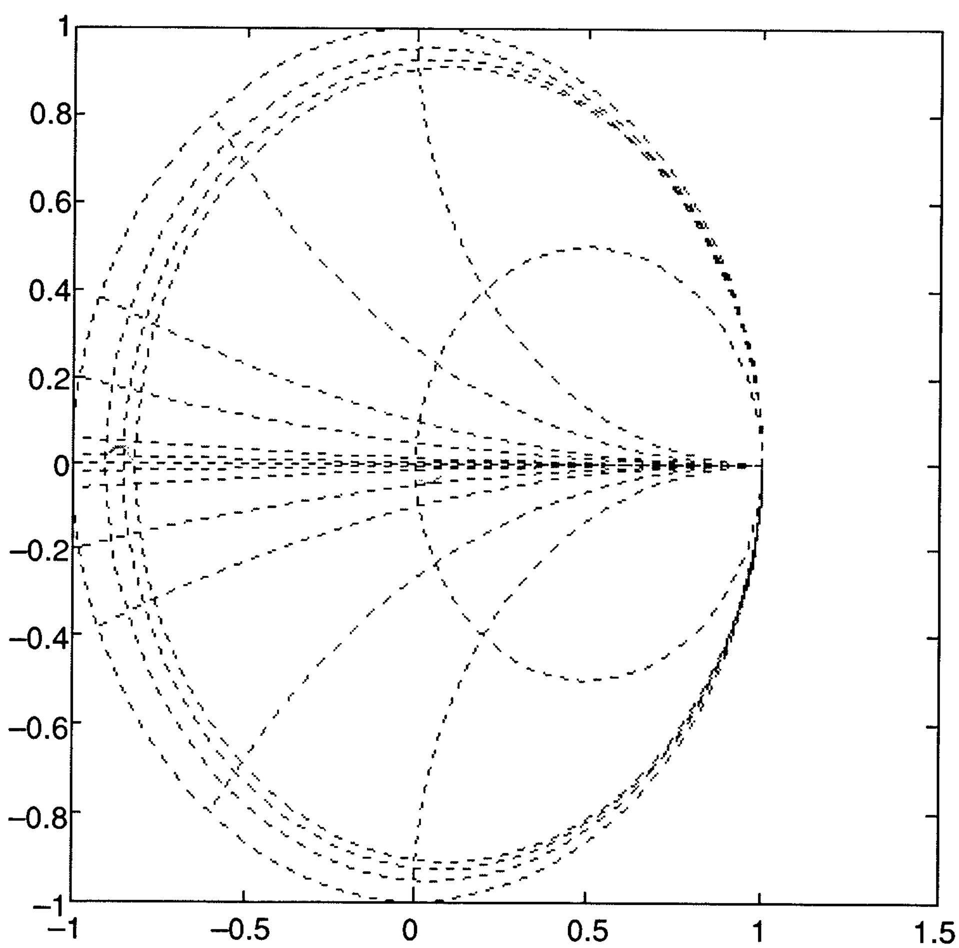


Figure 4. Locations of [S] on unit Smith Chart.



Listing of Code

```
%rspice.m
% swap of nodes k,l e.g. k=2, l=5
 % starting w/known Y(n,m)

Zo=50;pts=51; TRUE=1; % CONSTANTS
Y=[.2-j*.0398 j*.0398 0; j*.0398 1-j*.0105 -1; 0 -1 1.2];
Rg=0; Rs=0; Rd=0 ;Lg=0; Ls=0; Ld=0;
[n,m]=size(Y); k=2;L=3;

for FR=1:pts
   w=2*pi*1e9*FR; %m=1/w;
   ww(FR)=w;
   ymod(1,1,FR)=.2+(1./(w*i*1e-9));    ymod(1,2,FR)=-1./j*w*1e-9;                      ymod(1,3,FR)=0;
   ymod(2,1,FR)=-1./j*w*1e-9;          ymod(2,2,FR)=(j*w*2e-12)+1+(1./j*w*1e-9);       ymod(2,3,FR)=-1;
   ymod(3,1,FR)=0;                     ymod(3,2,FR)=-1;                                ymod(3,3,FR)=1.2;
end

for row=1:n
  for column=1:m
     for FR=1:pts
          switch TRUE
                    case ( ((row~=k)&(row~=L)) & ((column~=k)&(column~=L))),   ymodP(row,column,FR)=ymod(row,column,FR);
                    case ( ((row~=k)&(row~=L)) & column==k),              ymodP(row,column,FR)=ymod(row,L,FR);
                    case ( ((row~=k)&(row~=L)) & column==L),              ymodP(row,column,FR)=ymod(row,k,FR);
                    case (row==k & column==L),                       ymodP(row,column,FR)=ymod(L,k,FR);
                    case (row==k & column==k),                       ymodP(row,column,FR)=ymod(L,L,FR);
                    case (row==k & ((column~=k)&(column~=L)) ),          ymodP(row,column,FR)=ymod(L,column,FR);
                    case (row==L & ((column~=k)&(column~=L)) ),          ymodP(row,column,FR)=ymod(k,column,FR);
                    case (row==L & column==k),                       ymodP(row,column,FR)=ymod(k,L,FR);
                    case (row==L & column==L),                       ymodP(row,column,FR)=ymod(k,k,FR);
          end
     end
  end
end

for FR=1:pts
   % reduce Y-matrix to just the external nodes (ports) using Householder's Theorem
   nn=2; mm=2;
   yee(:,:,FR)=ymodP(1:nn,1:mm,FR);
   yei(:,:,FR)=ymodP(1:nn,mm+1:m,FR);
   yie(:,:,FR)=ymodP(nn+1:n, 1:mm,FR);
   yii(:,:,FR)=ymodP(nn+1:n,mm+1:m,FR);
   yred(:,:,FR)=yee(:,:,FR)-(yei(:,:,FR)*inv(yii(:,:,FR))*yie(:,:,FR));

          %%%%%%%%%%%% convert to intrinsic, z-parameters of model %%%%%%%%
          det_y(FR) = yred(1,1,FR).*yred(2,2,FR) - yred(1,2,FR).*yred(2,1,FR);
          zmod(1,1,FR) =  yred(2,2,FR)./det_y(FR);             zmod(1,2,FR) = -yred(1,2,FR)./det_y(FR);
          zmod(2,1,FR) = -yred(2,1,FR)./det_y(FR);             zmod(2,2,FR) = yred(1,1,FR)./det_y(FR);
```

Listing of Code (cont'd)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%          7. Calculate extrinsic,Z, parameters  by adding parasitics back to zmod          %%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        Zmod(1,1,FR)=zmod(1,1,FR) + (Rg+Rs) + j.*ww(FR).*(Lg + Ls);
        Zmod(1,2,FR)=zmod(1,2,FR) +    Rs  - j.*ww(FR).*Ls;
        Zmod(2,1,FR)=zmod(2,1,FR) +    Rs  + j.*ww(FR).*Ls;
        Zmod(2,2,FR)=zmod(2,2,FR) + (Rd+Rs) + j.*ww(FR).*(Ld + Ls);

        ZmodP(1,1,FR)=Zmod(1,1,FR)./Zo;  ZmodP(1,2,FR)=Zmod(1,2,FR)./Zo;
        ZmodP(2,1,FR)=Zmod(2,1,FR)./Zo;  ZmodP(2,2,FR)=Zmod(2,2,FR)./Zo;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%          8. Convert Zmod to S_mod          %%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        delt1(FR)=(ZmodP(1,1,FR)+1).*(ZmodP(2,2,FR)+1) -  (ZmodP(1,2,FR).*ZmodP(2,1,FR));
        Smod(1,1,FR)=( (ZmodP(1,1,FR)-1).*(ZmodP(2,2,FR)+1) - (ZmodP(1,2,FR).*ZmodP(2,1,FR)) )./delt1(FR);
        Smod(1,2,FR)=2.*ZmodP(1,2,FR)./delt1(FR);
        Smod(2,1,FR)=2.*ZmodP(2,1,FR)./delt1(FR);
        Smod(2,2,FR)=((ZmodP(1,1,FR)+1).*(ZmodP(2,2,FR)-1) - (ZmodP(1,2,FR)).*(ZmodP(2,1,FR)))./delt1(FR);

  Smod11(FR)=Smod(1,1,FR);  Smod12(FR)=Smod(1,2,FR);   Smod21(FR)=Smod(2,1,FR);   Smod22(FR)=Smod(2,2,FR);

end
```

Conclusion/Recommendations

Improvements that could be made include a text-based input interface similar to a SPICE net list or even a graphical user input. As implemented, it can easily be modified to be included in larger MATLAB programs, e.g., to simulate an empirical (circuit based) model of a transistor in order to compare it with measured data.

References

[1] Microwave Transistor Amplifiers, Gonzalez.

[2] Microwave & RF Circuits: Analysis, Design, Fabrication & Measurement, Notes from M.L. Edwards (1995).

[3] Advanced Engineering Mathematics, 7th Edition, Kreyszig.

Distribution

Admnstr
Defns Techl Info Ctr
ATTN DTIC-OCP
8725 John J Kingman Rd Ste 0944
FT Belvoir VA 22060-6218

US Army Rsrch Lab
ATTN AMSRL-CI-IS-R Mail & Records Mgmt
ATTN AMSRL-CI-IS-T Techl Pub (2 copies)
ATTN AMSRL-CI-OK-TL Techl Lib (2 copies)
ATTN AMSRL-D D R Smith

US Army Rsrch Lab (cont'd)
ATTN AMSRL-DD J M Miller
ATTN AMSRL-SE-EP M Delancey
ATTN AMSRL-SE-RE C Fazi
ATTN AMSRL-SE-RE F Crowne
ATTN AMSRL-SE-RE M Patterson
ATTN AMSRL-SE-RE R del Rosario (5 copies)
ATTN AMSRL-SE-RL S Svensson
Adelphi MD 20783-1197

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE March 2002

3. REPORT TYPE AND DATES COVERED Progress, Aug 01 to Sept 01

4. TITLE AND SUBTITLE Microwave Circuit Simulator for MATLAB

5. FUNDING NUMBERS
DA PR: AH94
PE: 62705A

6. AUTHOR(S) Romeo D. del Rosario and Daniel C. Judy

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
Attn: AMSRL-SE-RE email: romeo@arl.army.mil
2800 Powder Mill Road
Adelphi, MD 20783-1197

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TN-184

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
2800 Powder Mill Road
Adelphi, MD 20783-1197

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
ARL PR: 2NE6K1
AMS code: 622705.H9411

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
This technical note describes a microwave circuit simulator implemented in the technical computing language, MATLAB. The simulator takes an input circuit with a known admittance matrix, performs a node swap that puts the external nodes (ports) at the upper left hand of an equivalent admittance matrix, then reduces the matrix (regardless of size) to include only the external nodes using Householder's method. The resulting 2×2 admittance matrix provides all the information to uniquely define the circuit from its external ports and facilitates straightforward calculation of Z-parameters, S-parameters, etc. (In our case, the reduced matrix is 2×2; however, the method applies to multiple port devices as well.) Although it can be used by itself, the simulator is designed with a bent toward empirical (circuit based) transistor models and may be incorporated into small and large signal transistor models.

14. SUBJECT TERMS microwave circuit simulation

15. NUMBER OF PAGES 16

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102